



# UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE  
United States Patent and Trademark Office  
Address: COMMISSIONER FOR PATENTS  
P.O. Box 1450  
Alexandria, Virginia 22313-1450  
www.uspto.gov

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
09/839,045	04/20/2001	Dietrich Charisius	30013630-0008	3644

23485 7590 05/22/2006

JINAN GLASGOW  
300 N. GREENE ST., SUITE 1600  
P.O. BOX 2974  
GREENSBORO, NC 27401

EXAMINER

INGBERG, TODD D

ART UNIT	PAPER NUMBER
----------	--------------

2193

DATE MAILED: 05/22/2006

Please find below and/or attached an Office communication concerning this application or proceeding.

## Office Action Summary

Application No.

09/839,045

Applicant(s)

CHARISIUS ET AL.

Examiner

Todd Ingberg

Art Unit

2193

-- The MAILING DATE of this communication appears on the cover sheet with the correspondence address --

### Period for Reply

A SHORTENED STATUTORY PERIOD FOR REPLY IS SET TO EXPIRE 3 MONTH(S) OR THIRTY (30) DAYS, WHICHEVER IS LONGER, FROM THE MAILING DATE OF THIS COMMUNICATION.

- Extensions of time may be available under the provisions of 37 CFR 1.136(a). In no event, however, may a reply be timely filed after SIX (6) MONTHS from the mailing date of this communication.
- If NO period for reply is specified above, the maximum statutory period will apply and will expire SIX (6) MONTHS from the mailing date of this communication.
- Failure to reply within the set or extended period for reply will, by statute, cause the application to become ABANDONED (35 U.S.C. § 133). Any reply received by the Office later than three months after the mailing date of this communication, even if timely filed, may reduce any earned patent term adjustment. See 37 CFR 1.704(b).

### Status

- 1) ☒ Responsive to communication(s) filed on 02 March 2006.
- 2a) ☐ This action is **FINAL**. 2b) ☒ This action is non-final.
- 3) ☐ Since this application is in condition for allowance except for formal matters, prosecution as to the merits is closed in accordance with the practice under *Ex parte Quayle*, 1935 C.D. 11, 453 O.G. 213.

### Disposition of Claims

- 4) ☒ Claim(s) See Continuation Sheet is/are pending in the application.
- 4a) Of the above claim(s) 2, 3, 11-19, 22, 24-25, 33-41, 43, 46, 53-67, 69-70, 73-74, 77-78, 80-82, 84-85, 93-101, 104, 106-107, 115-123, 125, 128, 135-149, 151-152, 155-156, 159-160, 162, 164, 166-167, 174-175 and 181 is/are withdrawn from consideration.
- 5) ☐ Claim(s) \_\_\_\_\_ is/are allowed.
- 6) ☒ Claim(s) See Continuation Sheet is/are rejected.
- 7) ☐ Claim(s) \_\_\_\_\_ is/are objected to.
- 8) ☐ Claim(s) \_\_\_\_\_ are subject to restriction and/or election requirement.

### Application Papers

- 9) ☐ The specification is objected to by the Examiner.
- 10) ☒ The drawing(s) filed on 20 April 2001 is/are: a) ☒ accepted or b) ☐ objected to by the Examiner.  
Applicant may not request that any objection to the drawing(s) be held in abeyance. See 37 CFR 1.85(a).  
Replacement drawing sheet(s) including the correction is required if the drawing(s) is objected to. See 37 CFR 1.121(d).
- 11) ☐ The oath or declaration is objected to by the Examiner. Note the attached Office Action or form PTO-152.

### Priority under 35 U.S.C. § 119

- 12) ☐ Acknowledgment is made of a claim for foreign priority under 35 U.S.C. § 119(a)-(d) or (f).
- a) ☐ All b) ☐ Some \* c) ☐ None of:
- ☐ Certified copies of the priority documents have been received.
  - ☐ Certified copies of the priority documents have been received in Application No. \_\_\_\_\_.
  - ☐ Copies of the certified copies of the priority documents have been received in this National Stage application from the International Bureau (PCT Rule 17.2(a)).
- \* See the attached detailed Office action for a list of the certified copies not received.

### Attachment(s)

- 1) ☒ Notice of References Cited (PTO-892)
- 2) ☐ Notice of Draftsperson's Patent Drawing Review (PTO-948)
- 3) ☐ Information Disclosure Statement(s) (PTO-1449 or PTO/SB/08)  
Paper No(s)/Mail Date \_\_\_\_\_
- 4) ☐ Interview Summary (PTO-413)  
Paper No(s)/Mail Date. \_\_\_\_\_
- 5) ☐ Notice of Informal Patent Application (PTO-152)
- 6) ☐ Other: \_\_\_\_\_

Continuation of Disposition of Claims: Claims pending in the application are 1,4-10,20,21,23,26-32,42,44,45,47-52,68,71,72,75,76,79,83,86-92,102,103,105,108-114,124,126,127,129-134,150,153,154,157,158,161,163,165,168-173,176-180 and 182-213.

Continuation of Disposition of Claims: Claims rejected are 1,4-10,20,21,23,26-32,42,44,45,47-52,68,71,72,75,76,79,83,86-92,102,103,105,108-114,124,126,127,129-134,150,153,154,157,158,161,163,165,168-173,176-180 and 182-213.

Art Unit: 2193

## **DETAILED ACTION**

### **Claim Status**

Claims 1,4-10,20-21,23,26-32,42,44-45,47-52,68,71-72,75-76,79,83,86-92,102-103,105,108-114,124, 126-127, 129-134,150, 153-154, 157-158,161,163,165,168-173,176-180 and 182-213 have been examined.

Claims 1, 20, 42, 68, 75, 83, 102, 124, 150, 157, 165, 171, 178, 180, 183, 187, 190, 191, 192 and 193 have been amended.

Claims 2, 3, 11-19, 22, 24-25, 33-41, 43, 46, 53-67, 69-70, 73-74, 77-78, 80-82, 84-85, 93-101, 104, 106-107, 115-123, 125, 128, 135-149, 151-152, 155-156, 159-160, 162, 164, 166-167, 174-175 and 181 are cancelled.

### ***Claim Rejections - 35 USC § 101***

1. 35 U.S.C. 101 reads as follows:

Whoever invents or discovers any new and useful process, machine, manufacture, or composition of matter, or any new and useful improvement thereof, may obtain a patent therefor, subject to the conditions and requirements of this title.

2. Claims 183, 190, 191, 192 and 193 are rejected under 35 U.S.C. § 101. The claimed invention is not producing a tangible result. These claims have the “means for displaying” but are not actually display. Displaying, updating, writing or storing to a computer readable medium produces a tangible result. The remaining independent claims are interpreted to be displaying on a computer monitor (tangible embodiment).

***Claim Rejections - 35 USC § 102***

3. The following is a quotation of the appropriate paragraphs of 35 U.S.C. 102 that form the basis for the rejections under this section made in this Office action:

A person shall be entitled to a patent unless –

(b) the invention was patented or described in a printed publication in this or a foreign country or in public use or on sale in this country, more than one year prior to the date of application for patent in the United States.

4. Claims 1,4-10,20-21,23,26-32,42,44-45,47-52,68,75,83,86-92,102-103,105,108-114,124, 126-127, 129-134,150, 157,163,165,168-173,176, 178 -181 and 183-188, 190-194, 197 – 199, 201- 203, 205 – 207, 209-211 and 213 are rejected under 35 U.S.C. 102(b) as being anticipated by “Principles of Object-Oriented Analysis and Design”, James Martin, June 1, 1992.

**Claim 1**

A method in a data processing system having a plurality of elements, each element having corresponding code and a graphical representation, the method comprising the steps of displaying the graphical representation of the corresponding code of each of the plurality of elements including a first element and second element; receiving a request to form a link; receiving an indication of a first of the plurality of elements; receiving an indication of a second of the plurality of elements; and in response to receiving the request, the indication of the first element, and the indication of the second element, generating new code independent of the graphical representation and adding the new code to the first element to reflect the link to the second element and modifying the graphical representation of the code associated with the first element to reflect the link to the second element.

**Rejection for Claim**

Martin anticipates a modeling system where elements (Martin, pages 52 – 56, 142 – 143, 155 – 197, 226 – 229,269 – 290) diagrams are executable – diagrams consisting of - classes, objects etc). The corresponding code and a graphical representation are displayed. The graphical representations are executable thus the relationship between the code and the diagrams is linked. Change in either the Graphical or textual design affect each other (Martin, pages 155 and 310). In looking at class and object modeling (Martin, Chapters 5, 7 and 18). The diagrams are language neutral (Martin, Appendix A). The principle of inheritance is implemented (Martin, pages 24 – 25, 30, 101, 174 – 175, 266 – 269, 315 and 389-390). Note : source and destination are referred to as child/parent or base/derived by most OO authors terminology. The relationships of among classes and objects are definable (Martin, chapter 7). Nested code is also generated by the tool (Martin, pages 225 – 229). The Instant CASE tool employs executable diagrams, to generate code (Martin, page 7, code generator is driven by other parts of the tool) an interpreter and an optimizing compiler are employed (Martin, pages 285 - 299).

**Claim 4**

The method of claim 1, wherein the step of adding new code to the first element comprises the steps of determining whether linking the first element to the second element would violate a predefined rule; and when it is determined that linking the first element to the second element

Art Unit: 2193

would not violate a predefined rule, adding the new code to the first element to form the link to the second element.

**Rejection for Claim**

As per the rejection for claim 1 and the modeling of Martin teaches built in validation (Martin, page 53 and 228). Code is added or removed based on the user interaction with the executable model and the passing of the validation as described in the section above.

**Claim 5**

The method of claim 4, wherein the step of determining whether linking the first element to the second element would violate a predefined rule comprises the steps of: determining whether the first element is a class and whether the second element is another class; and when it is determined that the first element is the class and that the second element is the other class, identifying the link from the first element to the second element as an inheritance link.

**Rejection for Claim**

See the rejection for claim 1 (Inheritance) and claim 4 (syntax checking in modeling).

**Claim 6**

The method of claim 5, further comprising the step of identifying a link error when it is determined that the first element is the class and that the second element is not the other class.

**Rejection for Claim**

See the rejection for claim 4.

**Claim 7**

The method of claim 4 wherein the step of determining whether linking the first element to the second element would violate a predefined rule comprises the steps of: determining whether the first element is a class and whether the second element is an interface; and when it is determined that the first element is the class and the second element is the interface, identifying the link from the first element to the second element as an implementation link.

**Rejection for Claim**

See the rejection for claim 4 and claim 1 types of relationships.

**Claim 8**

The method of claim 7, further comprising the step of identifying a link error when it is determined that the first element is the class and that the second element is not the interface.

**Rejection for Claim**

See the rejection for claim 4.

**Claim 9**

The method of claim 4, wherein the step of determining whether linking the first element to the second element would violate a predefined rule comprises the steps of: determining whether the first element is an interface and the second element is another interface; and when it is determined that the first element is the interface and the second element is the other interface, identifying the link from the first element to the second element as an inheritance link.

Art Unit: 2193

**Rejection for Claim**

See the rejection for claim 5.

**Claim 10**

The method of claim 9, further comprising the step of identifying a link error when it is determined that the first element is the interface and the second element is not the other interface.

**Rejection for Claim**

See the rejection for claim 4.

**Claim 20**

A method in a data processing system having a plurality of elements and having a link between two of the plurality of elements, wherein each element has corresponding code along with a graphical representation and the linked elements include it source and a destination, the method comprising the steps of displaying the graphical representation of the corresponding code of each of the plurality of elements including the source and the destination; receiving an identification of the link; receiving a selection of one of the linked elements; receiving an identification of another of the plurality of elements that is different than the linked elements, wherein a graphical representation of the corresponding code of the other element is displayed; determining whether the selected element is the destination; and when it is determined that the selected element is the destination, modifying the corresponding code of the other element independently of the graphical representation in order to reflect a new link between the other element and the destination element and modifying the graphical representation of the corresponding code of the other element to reflect the new link between the other element and the destination element.

**Rejection for Claim**

See the rejection for claim 1.

**Claim 21**

The method of claim 20, wherein the modifying step further include the step of modifying the corresponding code of the source to reflect the removal of the link between the source and the destination.

**Rejection for Claim**

The diagrams are executable and the alteration of the graphical model changes the code model and vice versa (Martin, page 52 and page 269, canceling inheritance).

**Claim 26**

The method of claim 20, wherein the modifying step includes the steps of determining whether linking the other element to the destination would violate a predefined rule; and when it is determined that linking the other element to the destination would not violate a predefined rule, modifying the corresponding code of the source to reflect the removal of the link between the source and the destination; and adding new code to the corresponding code of the other element to reflect the new link between the other element and the destination element.

**Rejection for Claim**

See the rejections for claim 20 and claim 4.

Art Unit: 2193

**Claim 27**

The method of claim 26, wherein the step of determining whether linking the other element to the destination would violate a predefined rule, comprises the steps of: determining whether the other element is a class and whether the destination is another class; and when it is determined that the other element is the class and that the destination is the other class, identifying the new link between the other element and the destination as an inheritance link.

**Rejection for Claim**

See the rejections for claim 4 and claim 5.

**Claim 28**

The method of claim 26, wherein the step of determining whether linking the other element to the destination would violate a predefined rule, comprises the steps of: determining whether the other element is a class and whether the destination is an interface; and when it is determined that the other element is the class and that the destination is the interface, identifying the new link between the other element and the destination as an implementation link.

**Rejection for Claim**

See the rejections for claim 21 (object modeling) and claim 4.

**Claim 29**

The method of claim 28, further comprising the step of identifying a link error when it is determined that the other element is the class and that the destination is not the interface.

**Rejection for Claim**

See the rejections for claim 4

**Claim 30**

The method of claim 26, wherein the step of determining whether linking the other element to the destination would violate a predefined rule, comprises the steps of: determining whether the other element is an interface and the destination is another interface; and when it is determined that the other element is the interface and the destination is the other interface, identifying the new link between the other element and the destination as an inheritance link.

**Rejection for Claim**

See the rejections for claim 5.

**Claim 31**

The method of claim 30, further comprising the step of identifying a link error when it is determined that the other element is not the interface.

**Rejection for Claim**

See the rejections for claim 4

**Claim 32**

The method of claim 30, further comprising the step of identifying a link error when it is determined that the destination is not the other interface.

**Rejection for Claim**

See the rejections for claim 4



Art Unit: 2193

**Claim 42**

A method in a data processing system having a plurality of elements and having a link between two of the plurality of elements, wherein each element has corresponding code along with a graphical representation and the linked elements include a source and a destination, the method comprising the steps of: displaying the graphical representation of the corresponding code of each of the plurality of elements including the source and destination; receiving an identification of the link; receiving a selection of one of the linked elements; receiving an identification of another of the plurality of elements that is different than the linked elements, wherein a graphical representation of the corresponding code of the other element is displayed; determining whether the selected element is the source and when it is determined that the selected element is the source, modifying the corresponding code of the source independently of the graphical representation in order to reflect a new link between the source and the other element and modifying the graphical representation of the corresponding code of the source element to reflect the new link between the source element and the other element.

**Rejection for Claim**

See the rejections for claim 1 and claim 20

**Claim 45**

The method of claim 42, further comprising the steps of: when it is determined that the selected element is the source, determining whether linking the source to the other element would violate a predefined rule; and when it is determined that linking the source to the other element would not violate a predefined rule, modifying the corresponding code of the source to reflect the removal of the link between the source and the destination; and adding new code to the corresponding code of the source to reflect the new link to the other element.

**Rejection for Claim**

See the rejections for claim 20, claim 4 and claim 21,

**Claim 47**

The method of claim 45, wherein the step of determining whether linking the source to the other element would violate a predefined rule, comprises the steps of: determining whether the source is a class and whether the other element is another class; and when it is determined that the source is the class and the other element is the other class, identifying the new link between the source and the other element as an inheritance link.

**Rejection for Claim**

See the rejections for claim 4 and claim 5

**Claim 51**

The method of claim 50, further comprising the step of identifying a link error when it is determined that the source is not the interface.

**Rejection for Claim**

See the rejections for claim 4.

Art Unit: 2193

**Claim 52**

The method of claim 50, further comprising the step of identifying a link error when it is determined that the other element is not the other interface.

**Rejection for Claim**

See the rejections for claim 4.

**Claim 184**

The method of claim 42, further comprising the step of modifying the code corresponding to the source to reflect the removal of the link to the destination.

**Rejection for Claim**

See the rejections for claim 21.

**Claim 23**

The method of claim 184, further comprising the step of modifying the graphical representation of the corresponding code of the source to reflect the removal of the link between the source and the destination.

**Rejection for Claim**

See the rejections for claim 21.

**Claim 44**

The method of claim 184, further comprising the step of modifying the graphical representation of the code corresponding the source to reflect the removal of the link to the destination.

**Rejection for Claim**

See the rejections for claim 21.

**Claim 68**

A method in a data processing system having a plurality of elements having a graphical representation, the method comprising the steps of:

- displaying the graphical representation of the corresponding code of each of the plurality of elements including s first and second element;

- receiving an identification of a first of the plurality of elements;

- receiving an identification of a second of the plurality of elements;

- receiving an indication that the first element is to be included in the second element;

- determining whether the inclusion of the first element in the second element would violate a predefined rule; and

when it is determined that the inclusion of the first element in the second element would not violate a predefined rule, transferring the code corresponding to the first element into the second element, wherein said code transfer occurs independently of the graphical representation and modifying the graphical representation of the code of the second element to reflect the transfer of the code corresponding to the first element into the second element.

**Rejection for Claim**

See the rejections for claim 1 and claim 4.

Art Unit: 2193

**Claim 75**

A method in a data processing system having a plurality of elements, each element having corresponding code and a graphical representation wherein code corresponding to a first of the plurality of elements is nested in the code corresponding to second of the plurality of elements, the method comprising the steps of: displaying the graphical representation of the corresponding code of each of the plurality of elements including the first element and the second element; receiving an indication that the first element is to be removed from the second element; determining whether the removal of the first element from the second element would not violate a predefined rule; and when it is determined that the removal of the first element from the second element would not violate a predefined rule, removing code corresponding to the first element from the second element, wherein said code removal occurs independently of the graphical representation and modifying the graphical representation of the code corresponding to the second element to reflect the removal of the first element from the second element.

**Rejection for Claim**

See the rejections for claim 1 and claim 4.

**Claim 83**

A computer-readable medium containing instructions for controlling a data processing system to perform a method, the data processing system having a plurality of elements, each element having corresponding code and a graphical representation, the method comprising the steps of: displaying the graphical representation of the corresponding code of the plurality of elements including the first element and the second element; receiving a request to form a link; receiving an indication of a first of the plurality of elements; receiving an indication of a second of the plurality of elements; and in response to receiving the request, the indication of the first element, and the indication of the second element, generating new code independent of the graphical representation and adding the new code to the first element to reflect the link to the second element and modifying the graphical representation of the code associated with the first element to reflect the link to the second element.

**Rejection for Claim**

See the rejections for claim 1

**Claim 86**

The computer-readable medium of claim 83, wherein the step of adding new code to the first element comprises the steps of: determining whether linking the first element to the second element would violate a predefined rule; and when it is determined that linking the first element to the second element would not violate a predefined rule, adding the new code to the first element to form the link to the second element.

**Rejection for Claim**

See the rejections for claim 4.

**Claim 87**

The computer-readable medium of claim 86, wherein the step of determining whether linking the first element to the second element would violate a predefined rule comprises the steps of: determining whether the first element is a class and whether the second element is another class;

Art Unit: 2193

and when it is determined that the first element is the class and that the second element is the other class, identifying the link from the first element to the second element as an inheritance link.

**Rejection for Claim**

See the rejections for claim 4.

**Claim 88**

The computer-readable medium of claim 87, wherein the method further comprises the step of identifying a link error when it is determined that the first element is the class and that the second element is not the other class.

**Rejection for Claim**

See the rejections for claim 4.

**Claim 89**

The computer-readable medium of claim 86, wherein the step of determining whether linking the first element to the second element would violate a predefined rule comprises the steps of: determining whether the first element is a class and whether the second element is an interface; and when it is determined that the first element is the class and that the second element is the interface, identifying the link from the first element to the second element as an implementation link.

**Rejection for Claim**

See the rejections for claim 7.

**Claim 90**

The computer-readable medium of claim 89, wherein the method further comprises the step of identifying a link error when it is determined that the first element is the class and that the second element is not the interface.

**Rejection for Claim**

See the rejections for claim 4.

**Claim 91**

The computer-readable medium of claim 86, wherein the step of determining whether linking the first element to the second element would violate a predefined rule comprises the steps of: determining whether the first element is an interface and the second element is another interface; and when it is determined that the first element is the interface and the second element is the other interface, identifying the link from the first element to the second element as an inheritance link.

**Rejection for Claim**

See the rejections for claim 5.

**Claim 92**

The computer-readable medium of claim 91, wherein the method further comprises the step of identifying a link error when it is determined that the first element is the interface and the second element is not the other interface.

Art Unit: 2193

**Rejection for Claim**

See the rejections for claim 4.

**Claim 102**

A computer-readable medium containing instructions for controlling a data processing system to perform a method, the data processing system having a plurality of elements and having a link between two of the plurality of elements, wherein each element has corresponding code along with a graphical representation and the linked elements include a source and a destination, the method comprising the steps of displaying a graphical representation of the corresponding code of each of the plurality of elements including the source and destination; receiving an identification of the link; receiving a selection of one of the linked elements; receiving an identification of another of the plurality of elements that is different than the linked elements, wherein a graphical representation of the corresponding code of the other element is displayed; determining whether the selected element is the destination; and when it is determined that the selected element is the destination, modifying the corresponding code of the other element independently of the graphical representation in order to reflect a new link between the other element and the destination element and modifying the graphical representation of the corresponding code of the other element to reflect the new link between the other element and the destination element.

**Rejection for Claim**

See the rejections for claim 20.

**Claim 103**

The computer-readable medium of claim 102, wherein the modifying step further includes the step of modifying the corresponding code of the source to reflect the removal of the link between the source and the destination.

**Rejection for Claim**

See the rejections for claim 20 and claim 21.

**Claim 105**

The computer-readable medium of claim 102, wherein the method further comprises the step of modifying the graphical representation of the corresponding code of the source to reflect the removal of the link between the source and the destination.

**Rejection for Claim**

See the rejections for claim 21.

**Claim 108**

The computer-readable medium of claim 102, wherein the modifying step includes the steps of determining whether linking the other element to the destination would violate a predefined rule; and when it is determined that linking the other element to the destination would not violate a predefined rule, modifying the corresponding code of the source to reflect the removal of the link between the source and the destination; and adding new code to the corresponding code of the other element to reflect the new link between the other element and the destination element.

**Rejection for Claim**

Art Unit: 2193

See the rejections for claim 4.

**Claim 109**

The computer-readable medium of claim 108, wherein the step of determining whether linking the other element to the destination would violate a predefined rule, comprises the steps of: determining whether the other element is a class and whether the destination is another class; and when it is determined that the other element is the class and that the destination is the other class, identifying the new link between the other element and the destination as an inheritance link.

**Rejection for Claim**

See the rejections for claim 4 and claim 5.

**Claim 110**

The computer-readable medium of claim 108, wherein the step of determining whether linking the other element to the destination would violate a predefined rule, comprises the steps of: determining whether the other element is a class and whether the destination is an interface; and when it is determined that the other element is the class and chat the destination is the interface, identifying the new link between the other element and the destination as an implementation link.

**Rejection for Claim**

See the rejections for claim 4.

**Claim 111**

The computer-readable medium of claim 110 wherein the method further comprises the step of identifying a link error when it is determined that the other element is the class rend that the destination is not the interface.

**Rejection for Claim**

See the rejections for claim 4.

**Claim 112**

The computer-readable medium of claim 108, wherein the step of determining whether linking the other element to the destination would violate a predefined rule, comprises the steps of: determining whether the other element is an interface and the destination is another interface; and when it is determined that the other element is the interface and the destination is the other interface, identifying the new link between the other element and the destination as an inheritance link.

**Rejection for Claim**

See the rejections for claim 4 and claim 5.

**Claim 113**

The computer-readable medium of claim 112, wherein the method further comprises the step of identifying a link error when it is determined that the other element is not the interface.

**Rejection for Claim**

See the rejections for claim 4.

Art Unit: 2193

**Claim 114**

The computer-readable medium of claim 112, wherein the method further comprises the step of identifying a link error when it is determined that the destination is not the other interface.

**Rejection for Claim**

See the rejections for claim 4.

**Claim 124**

A computer-readable medium containing instructions for controlling a data processing system to perform a method, the data processing system having a plurality of elements and having a link between two of the plurality of elements, wherein each element has corresponding code along with a graphical representation and the linked elements include a source and a destination, the method comprising the steps of displaying the graphical representation of the corresponding code of each of the plurality of elements including the source and destination; receiving an identification of the link; receiving a selection of one of the linked elements; receiving an identification of another of the plurality of elements that is different than the linked elements, wherein a graphical representation of the corresponding code of the other element is displayed; determining whether the selected element is the source; and when it is determined that the selected element is the source, modifying the corresponding code of the source independent of the graphical representation in order to reflect a new link between the source and the other element and modifying the graphical representation of the corresponding code of the source to reflect the new link between the source and the other element.

**Rejection for Claim**

See the rejections for claim 1.

**Claim 127**

The computer-readable medium of claim 124, wherein the method further comprises the steps of: when it is determined that the selected element is the source, determining whether linking the source to the other element would violate a predefined rule; and when it is determined that linking the source to the other element would not violate a predefined rule, modifying the corresponding code of the source to reflect the removal of the link between the source and the destination; and adding new code to the corresponding code of the source to reflect the new link to the other element.

**Rejection for Claim**

See the rejections for claim 4 and claim 21.

**Claim 129**

The computer-readable medium of claim 127, wherein the step of determining whether linking the source to the other element would violate a predefined rule, comprises the steps of determining whether the source is a class and whether the other element is another class; and when it is determined that the source is the class and that the other element is the other class, identifying the new link between the source and the other element as an inheritance link.

**Rejection for Claim**

See the rejections for claim 4 and claim 5.

Art Unit: 2193

**Claim 130**

The computer-readable medium of claim 127, wherein the step of determining whether linking the source to the other element would violate a predefined rule, comprises the steps of determining whether the source is a class and whether the other element is an interface; and when it is determined that the source is the class and that the other element is the interface, identifying the new link from the source to the other element as an implementation link.

**Rejection for Claim**

See the rejections for claim 4 and claim 5.

**Claim 131**

The computer-readable medium of claim 130, wherein the method further comprises the step of identifying a link error when it is determined that the other element is not the interface.

**Rejection for Claim**

See the rejections for claim 4.

**Claim 132**

The computer-readable medium of claim 127, wherein the step of determining whether linking the source to the other element would violate a predefined rule, comprises the steps of: determining whether the source is an interface and the other element is another interface; and when it is determined that the source is the interface and the other element is the other interface, identifying the new link between the source and the other element as an inheritance link.

**Rejection for Claim**

See the rejections for claim 4.

**Claim 133**

The computer-readable medium of claim 132, wherein the method further comprises the step of identifying a link error when it is determined that the source is not the interface.

**Rejection for Claim**

See the rejections for claim 4.

**Claim 134**

The computer-readable medium of claim 132, wherein the method further comprises the step of identifying a link error when it is determined that the other element is not the other interface.

**Rejection for Claim**

See the rejections for claim 4.

**Claim 185**

The computer-readable medium of claim 124, wherein the method further comprises the step of modifying the code corresponding to the source to reflect the removal of the link to the destination

**Rejection for Claim**

See the rejections for claim 21.



Art Unit: 2193

**Claim 126**

The computer-readable medium of claim 185, wherein the method further comprises the step of modifying the graphical representation of the code corresponding to the source to reflect the removal of the link to the destination.

**Rejection for Claim**

See the rejections for claim 21.

**Claim 150**

A computer-readable medium containing instructions for controlling a data processing system to perform a method, the data processing system having a plurality of elements having a graphical representation, the method comprising the steps of: displaying the graphical representation of the corresponding code of each of the plurality of elements including a first and second element; receiving an identification of a first of the plurality of elements; receiving an identification of a second of the plurality of elements; receiving an indication that the first element is to be included in the second element; determining whether the inclusion of the first element in the second element would violate a predefined rule; and when it is determined that the inclusion of the first element in the second element would not violate a predefined rule, transferring the code corresponding to the first element into the second element; wherein the code transfer occurs independently of the graphical representation and modifying; the graphical representation of the code of the second element to reflect the transfer of the code corresponding to the first element into the second element.

**Rejection for Claim**

See the rejections for claim 1 and claim 4.

**Claim 157**

A computer-readable medium containing instructions for controlling a data processing system to perform a method, the data processing system having a plurality of elements, each element having corresponding code and a graphical representation wherein code corresponding to a first of the plurality of elements is nested in the code corresponding to a second of the plurality of elements, the method comprising the steps of: displaying the graphical representation of the corresponding code of each of the plurality of elements including the first element and the second element; receiving an indication that the first element is to be removed from the second element; determining whether the removal of the first element from the second element would violate a predefined rule; and when it is determined that the removal of the first element from the second element would not violate a predefined rule, removing the code corresponding to the first element from the second element wherein the code removal occurs independently of the graphical representation and modifying a graphical representation of the code corresponding to the second element to reflect the removal of the first element from the second element.

**Rejection for Claim**

See the rejections for claim 1 claim 4 and claim 21.

**Claim 165**

A data processing system comprising: a secondary storage device further comprising a plurality of elements, each element having corresponding code and a graphical representation; a memory

Art Unit: 2193

device further comprising a program that displays the graphical representation of the corresponding code of each of the plurality of elements including a first element and a second element, that receives a request to form a link that receives an indication of a first of the plurality of elements, that receives an indication of a second of the plurality of elements, that determines whether linking the first element to the second element would violate a predefined rule, that generates new code independent of the graphical representation and adds the new code to the first element to reflect the link to the second element when it is determined that linking the first element to the second element would not violate a predefined rule, and that modifies the graphical representation of the code associated with the first element to reflect the link to the second element; and a processor for running the program.

**Rejection for Claim**

See the rejections for claim 1 and claim 4.

**Claim 168**

The data processing system of claim 165, wherein when the program determines whether linking the first element to the second element would violate a predefined rule, the program determines whether the first element is a class and whether the second element is another class, and when it is determined that the first element is the class and that the second element is the other class, the program identifies the link from the first element to the second element as an inheritance link.

**Rejection for Claim**

See the rejections for claim 4 and claim 5.

**Claim 169**

The data processing system of claim 165, wherein when the program determines whether linking the first element to the second element would violate a predefined rule, the program determines whether the first element is a class and whether the second element is an interface, and when it is determined that the first element is the class and that the second element is the interface, the program identifies the link from the first element to the second element as an implementation.

**Rejection for Claim**

See the rejections for claim 7.

**Claim 170**

The data processing system of claim 165, wherein when the program determines whether linking the first element to second element would violate a predefined rule, the program determines whether the first element is an interface and the second element is another interface, and when it is determined that the first element is the interface and the second element is the other interface, the program identifies the link from the first element to the second element as an inheritance link.

**Rejection for Claim**

See the rejections for claim 5.

**Claim 194**

The data processing system in claim 165, further comprising language neutral representation of the source code, wherein the language neutral representation of the source code is used to

Art Unit: 2193

generate a graphical representation of the source code and textual representation of the source code.

**Rejection for Claim**

See the rejections for claim 1.

**Claim 195**

The data processing system in the claim 194, wherein modifications to the textual representation of the source code will immediately modify the corresponding graphical representation of the source code and vice versa, thereby allowing simultaneous viewing of the textual and graphical representations of the source code.

**Rejection for Claim**

See the rejections for claim 1.

**Claim 197**

The data processing system in claim 165, wherein the source code is used directly to generate a graphical representation of the source code and a textual representation of the source code.

**Rejection for Claim**

See the rejections for claim 1.

**Claim 199**

The data processing in claim 197, wherein modifications to the textual representation of the source code will immediately modify the corresponding graphical representation of the source code and vice versa, thereby allowing simultaneous viewing of the textual and graphical representation of the source code.

**Rejection for Claim**

See the rejections for claim 1.

**Claim 171**

A data processing system comprising: a secondary storage device further comprising a plurality of elements and having a link between two of the plurality of elements, wherein each element has corresponding code with a graphical representation and the linked elements include a source and a destination; a memory device further comprising a program that displays the graphical representation of the corresponding code of each of the plurality of elements including the source and the destination, that receives a selection of one of the linked elements, that receives an identification of another of the plurality of elements that is different than the linked elements, wherein a graphical representation of the corresponding code of the other element is

displayed, that determines whether the selected element is the destination, and that when it is determined that the selected element is the destination, generates new code independently of the graphical representation and adds new code to the code corresponding to the other element to reflect the new link between the other element and the destination when it is determined that the selected element is the destination, removes a portion of the corresponding code of the source that reflects the link between the source and the destination, modifies the graphical representation of the corresponding code of the source to reflect the removal of the link to the

Art Unit: 2193

destination, and modifies the graphical representation of the corresponding code of the other element to reflect the new link; and a processor for running the program.

**Rejection for Claim**

See the rejections for claim 1.

**Claim 172**

The data processing system of claim 171, wherein when it is determined that the other element is the class and that the destination is not the other class, the program further determines whether the destination is an interface, and when it is determined that the destination is the interface, the program identifies the new link between the other element and the destination as an implementation link.

**Rejection for Claim**

See the rejections for claim 7.

**Claim 173**

A data processing system of claim 171, wherein when it is determined that the other element is not the class and that the destination is not the other class, the program further determines whether the other element is an interface and whether the destination is another interface, and when it is determined that the other element is the interface and that the destination is the other interface, the program identified a the new link between the other element and the destination as an inheritance link.

**Rejection for Claim**

See the rejections for claim 5.

**Claim 186**

The data processing system of claim 171, wherein when it is determined that the other element is a class and that the destination is another class, the program identifies the new link between the other element and the destination as an inheritance link.

**Rejection for Claim**

See the rejections for claim 5.

**Claim 198**

The data processing system in claim 171, further comprising a language-neutral representation of the source code, wherein the language neutral representation of the source code is used to generate a graphical representation of the source code and a textual representation of the source code.

**Rejection for Claim**

See the rejections for claim 1.

**Claim 201**

The data processing system in claim 171, wherein the source code is used directly to generate a graphical representation of the source code and a textual representation of the source code.

**Rejection for Claim**

See the rejections for claim 1.

Art Unit: 2193

**Claim 178**

A data processing system comprising: a secondary storage device further comprising a plurality of elements, each element having corresponding code and a graphical representation; a memory device further comprising a program that displays the graphical representation of the code of a first of the plurality of elements, that receives an indication that the first element is to be included in the second element, that determines whether inclusion of the first element in the second element would violate a predefined rule,

That transfers code corresponding to the first element into the second element when it is determined that the inclusion of the first element in the second element would not violate a predefined rule, wherein the code transfer occurs independently of the graphical representation, and that modifies a graphical representation of the code of the second element to reflect the transfer of the first element into the second element; and a processor for running the program.

**Rejection for Claim**

See the rejections for claim 1 and claim 4.

**Claim 179**

The data processing system of claim 178, wherein the program removes the code corresponding to the first element from a file, places the code corresponding to the first element within the code corresponding to the second element and deletes the file corresponding to the first element.

**Rejection for Claim**

See the rejections for claim 21.

**Claim 206**

The data processing system is claim 178, further comprising a language-neutral representation of the source code, wherein the language neutral representation of the source code is used to generate a graphical representation of the source code and a textual representation of the source code.

**Rejection for Claim**

See the rejections for claim 1.

**Claim 207**

The data processing system in claim 205, wherein modifications to the textual representation of the source code will immediately modify the corresponding graphical representation of the source code and vice versa, thereby allowing simultaneous viewing of the textual and graphical representations of the source code.

**Rejection for Claim**

See the rejections for claim 1.

**Claim 209**

The data processing system in claim 178, wherein the source code is used directly to generate a graphical representation of the source code and a textual representation of the source code.

**Rejection for Claim**

See the rejections for claim 1.

Art Unit: 2193

**Claim 180**

A data processing system comprising: a secondary storage device further comprising a plurality of elements having graphical representations, wherein a first of the plurality of elements is nested within a second of the plurality of elements ; a memory device further comprising a program that displays the graphical representation of the code of a first of the plurality of elements and graphical representation of the code of a second of the plurality of elements, that receives an indication that the first element is to be removed from the second element would violate a predefined rule, and that removes the code corresponding to the first element from the second element when it is determined that the removal of the first element from the second element would not violate a predefined rule, wherein the code removal occurs independently of each graphical representation, that modifies the graphical representation of the second element to reflect the removal of the first element from the second element into a file; and a processor for running the program.

**Rejection for Claim**

See the rejections for claim 1 and claim 4.

**Claim 210**

The data processing system of claim 180, further comprising a language-neutral representation of the source code, wherein the language neutral representation of the source code is used to generate a graphical representation of the source code and a textual representation of the source code.

**Rejection for Claim**

See the rejections for claim 1.

**Claim 211**

The data processing system of claim 210, wherein modifications to the textual representation of the source code will immediately modify the corresponding graphical representation of the source code and vice versa, thereby allowing simultaneous viewing of the textual and graphical representations of the source code.

**Rejection for Claim**

See the rejections for claim 1.

**Claim 213**

The data processing system in claim 180, wherein the source code is used directly to generate a graphical representation of the source code and a textual representation of the source code.

**Rejection for Claim**

See the rejections for claim 1.

**Claim 183**

A system having a plurality of elements, each element having corresponding code and a graphical representation, the system comprising: means for displaying the graphical representation of the corresponding code of each of the plurality of elements; means for receiving a request to form a link; means for receiving an indication of a first of the plurality of

Art Unit: 2193

elements; means for receiving an indication of a second of the plurality of elements; and means for generating new code independent of the graphical representation and adding the new code to the first element to reflect the link to the second element in response to receiving the request, the indication of the first element, and the indication of the second element; and means for modifying the graphical representation of the code associated with the first element to reflect the link to the second element.

**Rejection for Claim**

See the rejections for claim 1.

**Claim 187**

A data processing system comprising: a secondary storage device further comprising a plurality of elements and having a link between two of the plurality of elements, wherein each element has corresponding code and a graphical representation and the linked elements include a source and a destination; a memory device further comprising a program that displays the graphical representation of the corresponding code of each of the plurality of elements including the source and the destination, that receives a selection of one of the linked elements, that receives an identification of another of the plurality of elements that is different than the linked elements, wherein a graphical representation of the corresponding code of the other element is displayed

**Rejection for Claim**

See the rejections for claim 1.

**Claim 176**

The data processing system of claim 187, wherein when it is determined that the source is the class and that the other element is not the other class, the program Further determines whether the other element is in an interface, and when it is determined that the other element is the interface, the program identifies the new link between the source and the other element as an implementation link.

**Rejection for Claim**

See the rejections for claim 7.

**Claim 188**

The data processing system of claim 187, wherein when it is determined that the source is a class and that the other element is another class, the program identifies the new link between the source and the other elements as an inheritance link.

**Rejection for Claim**

See the rejections for claim 5.

**Claim 202**

The data processing system of claim 187, further comprising a language-neutral representation of the source code, wherein the language neutral representation of the source code is used to generate a graphical representation of the source code and a textual representation of the source code.

**Rejection for Claim**

See the rejections for claim 1.

Art Unit: 2193

**Claim 203**

The data processing system in claim 202, wherein modifications to the textual representation of the source code will immediately modify the corresponding graphical representation of the source code and vice versa, thereby allowing simultaneous viewing of the textual and graphical representations of the source code.

**Rejection for Claim**

See the rejections for claim 1.

**Claim 205**

The data processing in claim 187, wherein the source code is used directly to generate a graphical representation of the source code and a textual representation of the source code.

**Rejection for Claim**

See the rejections for claim 1.

**Claim 190**

A system having a plurality of elements and having a link between two of the plurality of elements, wherein each element has corresponding code and graphical representation and the linked elements include a source and a destination, the system comprising: means for displaying the graphical representation of the corresponding code of each of the plurality of elements including the source and the destination; means for receiving an identification of the link; means for receiving a selection of one of the linked elements; means for receiving an identification of another of the plurality of elements that is different than the linked elements, wherein a graphical representation of the corresponding code of the other element is displayed; means for determining whether the selected element is the destination; and means for modifying the corresponding code of the other element to reflect a new link between the other element and the destination elements. Wherein the code modification occurs independently of the graphical representation, and modifying the graphical representation of the corresponding code of the other element to reflect the new link between the other element and the destination element when it is determined that the selected element is the destination.

**Rejection for Claim**

See the rejections for claim 1.

**Claim 191**

A system having a plurality of elements and having a link between two of the plurality of elements, wherein each element has corresponding code graphical representation and the linked elements include a source and a destination, the system comprising: means for displaying the graphical representation of the corresponding code of each of the plurality of elements including the source and destination; means for receiving an identification of the link; means for receiving a selection of one of the linked elements; means for receiving an identification of another of the plurality of elements that is different than the linked elements, wherein a graphical representation of the corresponding code of the other element is displayed; means for: modifying the corresponding code of the source to reflect a new link between the source and the other element wherein the corresponding code modification occurs independently of the graphical



Art Unit: 2193

representation, and modifying the graphical representation of the corresponding code of the source to reflect the new link between the source and the other element, when it is determined that the selected element is the source.

**Rejection for Claim**

See the rejections for claim 1.

**Claim 192**

A system having a plurality of elements, each having a graphical representation the system comprising: means for displaying the graphical representation of the corresponding code of each of the plurality of elements including a first and a second element; means for receiving an identification of a first of a plurality of elements; means for receiving an identification of a second of the plurality of elements; means for receiving an indication that the first element is to be included in the second element; means for transferring code corresponding to the first into the second elements wherein the code transfer occurs independently of the graphical representation; and means for modifying the graphical representation of the code of the second element to reflect the transfer of the code corresponding to the first element into the second element.

**Rejection for Claim**

See the rejections for claim 1.

**Claim 193**

A system having a plurality of elements, each element having corresponding code and a graphical representation, wherein code corresponding to a first of the plurality of elements is nested in code corresponding to a second of the plurality of elements, the system comprising: means for displaying the graphical representation of the corresponding code of each of the plurality of elements including the first element and the second element; means for receiving an indication that the first element is removed from the second element means for removing the code corresponding to the first element from the second element wherein said code removal occurs independently of the graphical representation; and means for modifying a graphical representation of the code corresponding to the second element to reflect the removal of the first element from the second element.

**Rejection for Claim**

See the rejections for claim 1 and claim 21.

***Claim Rejections - 35 USC § 103***

5. The following is a quotation of 35 U.S.C. 103(a) which forms the basis for all

obviousness rejections set forth in this Office action:

(a) A patent may not be obtained though the invention is not identically disclosed or described as set forth in section 102 of this title, if the differences between the subject matter sought to be patented and the prior art are such that the subject matter as a whole would have been obvious at the time the invention was made to a person having ordinary skill in the art to which said subject matter pertains. Patentability shall not be negated by the manner in which the invention was made.

Art Unit: 2193

6. Claims 71, 72, 76, 79, 153, 154, 158, 161, 195, 196, 200, 208, 182, 189, 212, 177 and 204 are rejected under 35 U.S.C. 103(a) as being unpatentable over “Principles of Object-Oriented Analysis and Design”, James Martin, June 1, 1992 in view of implementing the concept of Reuse in Object Technology.

**Rejection for Claims 196, 200, 204 208, 212**

Martin teaches the concepts of CASE tools through out the text book. Martin teaches the concept of a code repository (Martin, pages 6 – 8). Martin teaches the implementation of software in a variety of programming languages (Martin, page 13). It would have been obvious to one of ordinary skill at the time of invention to combine the teaches of Martin on Software Repositories and multiple languages and implement a multiple programming language repository, because Object Oriented Technology lends itself to reuse (Martin, pages 39 – 40).

**Claim 196**

The data processing system in claim 194, wherein the language neutral representation of the source code is generated from a multiplicity of programming languages.

**Claim 200**

The data processing system in claim 197, wherein the language neutral representation of the source code is generated from a multiplicity of programming languages.

**Claim 204**

The data processing system in claim 202, wherein the language neutral representation of the source code is generated from a multiplicity of programming languages.

**Claim 208**

The data processing system in claim 206, wherein the language neutral representation of the source code is generated from a multiplicity of programming languages.

**Claim 212**

The data processing system in claim 210, wherein the language neutral representation of the source code is generated from a multiplicity of programming languages.

**Rejection for Claims 71, 72, 76, 79, 153, 154, 158, 161, 182, 189 and 177 ion view of Martin in further view of SNAP User's Guide Development Environment, Version 7, Template Software, 1995, pages 3-1 to 3-30**

Martin teaches the various modeling techniques and the interaction of the graphical code and the textual code (claim 1). Martin does not teach the actual implementation of class files and directory structures for the model. It is Template who implements a model and teaches the support in the commercial product from 1995. Therefore, it would have been obvious to one of ordinary skill in the art at the time of invention because the ability to support modeling with an underlying directory structure and functions makes executable models more modular and easier to distribute.

Art Unit: 2193

**Claim 71**

The method of claim 68, wherein the step of transferring code comprises the steps of: removing the code corresponding to the first element from a file; placing the code corresponding to the first element within the code corresponding to the second element; and deleting the file corresponding to the first element.

**Claim 72**

The method of claim 68, wherein the method further comprises the steps of: when it is determined that the first element is the class and that the second element is not another class, determining whether the second element is a package; and when it is determined that the second element is a package, moving a file that includes code corresponding to the first element to a directory associated with the second element.

**Claim 76**

The method of claim 75, further comprising the step of: placing the code corresponding to the first element into a file.

**Claim 79**

The method of claim 75, further comprising the steps of: when it is determined that the first element is a class and that the second element is not another class, determining whether the second element is a package; and when it is determined that the second element is a package, removing a first file that includes code corresponding to the first element from a directory associated with the second element and placing the first file in another directory.

**Claim 153**

The computer-readable medium of claim 150, wherein the step of transferring code comprises the steps of: removing the code corresponding to the first element from a file corresponding to the second element; placing the code corresponding to the first element within the code corresponding to the second element; and deleting the file corresponding to the first element.

**Claim 154**

The computer-readable medium of claim 150, wherein the method further comprises the steps of: when it is determined that the first element is the class and that the second element is not the other class, determining whether the second element is a package; and when it is determined that the second element is a package, moving a file that includes code corresponding to the first element to a directory associated with the second element.

**Claim 158**

The computer-readable medium of claim 157, wherein the method further comprises the step of placing the code corresponding to the first element into a file.

**Claim 161**

Art Unit: 2193

The computer-readable medium of claim 157, wherein the method further comprises the steps of when it is determined that the first element is the class and that the second element is not another class, determining whether the second element is a package; and when it is determined that the second element is a package, removing a first file that includes code corresponding to the first element from a directory associated with the second element and placing the first file in another directory.

**Claim 182**

The data processing system of claim 180, wherein the first element is a class and the second element is not another class, the program further determines whether the second element is a package, and when it is determined that the second element is a package, the program removes the first file corresponding to the first element from a directory associated with the second element.

**Claim 189**

The data processing system of claim 180, wherein the method further comprises the step of placing the code corresponding to the first element into a file.

**Claim 177**

The data processing system of claim 187, wherein when it is determined that the source is not the class and that the other element is not the other class, the program further determines whether the source is an interface and the other element is another interface, and when it is determined that the source is the interface and the other element is the other interface, the program identifies the new link between the source and the other element as an inheritance link.

a secondary storage device further comprising a plurality of elements having a graphical representations, wherein a first of the plurality of elements is nested within a second of the plurality of elements; a memory device further comprising a program that displays the graphical representation of the code of a first of the plurality of elements and a graphical representation of the code of a second of the plurality of elements, that receives an indication that the first element is to be removed from the second element, that determines whether the removal of the first element from the second element would violate a predefined rule, and that removes the code corresponding to the first element from the second element when it is determined that the removal of the first element from the second element would not violate a predefined rule, wherein the code removal occurs independently of each graphical representation, that modifies the graphical representation of the second element to reflect the removal of the first element from the second element, and that places the code corresponding to the first element into a file; and a processor for running the program.

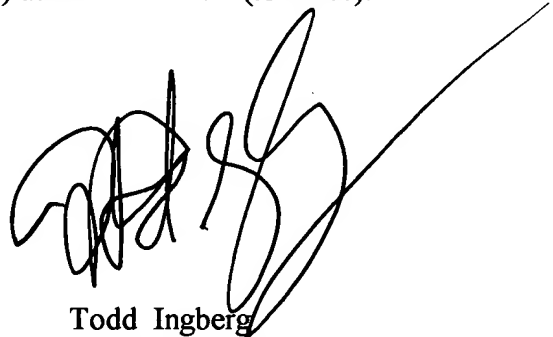
***Correspondence Information***

Art Unit: 2193

7. Any inquiry concerning this communication or earlier communications from the examiner should be directed to Todd Ingberg whose telephone number is (571) 272-3723. The examiner can normally be reached on during the work week..

If attempts to reach the examiner by telephone are unsuccessful, the examiner's supervisor, Kakali Chaki can be reached on (571) 272-3719. The fax phone number for the organization where this application or proceeding is assigned is 571-273-8300.

Information regarding the status of an application may be obtained from the Patent Application Information Retrieval (PAIR) system. Status information for published applications may be obtained from either Private PAIR or Public PAIR. Status information for unpublished applications is available through Private PAIR only. For more information about the PAIR system, see <http://pair-direct.uspto.gov>. Should you have questions on access to the Private PAIR system, contact the Electronic Business Center (EBC) at 866-217-9197 (toll-free).

A handwritten signature in black ink, appearing to be 'Todd Ingberg', written in a cursive, stylized script.

Todd Ingberg  
Primary Examiner  
Art Unit 2193

TI